Learning from Point Sets with Observational Bias

Liang Xiong
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213

Jeff Schneider
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Many objects can be represented as sets of multi-dimensional points. A common approach to learning from these point sets is to assume that each set is an *i.i.d.* sample from an unknown underlying distribution, and then estimate the similarities between these distributions. In realistic situations, however, the point sets are often subject to sampling biases due to variable or inconsistent observation actions. These biases can fundamentally change the observed distributions of points and distort the results of learning. In this paper we propose the use of conditional divergences to correct these distortions and learn from biased point sets effectively. Our empirical study shows that the proposed method can successfully correct the biases and achieve satisfactory learning performance.

1 INTRODUCTION

Traditional learning algorithms deal with fixed, finite dimensional vectors/points, but many real objects are actually sets of points that are multi-dimensional, real-valued vectors. For instance, in computer vision an image is often treated as a set of patches with each patch described by a fixed length feature vector (Li and Perona, 2005). In monitoring problems, each sensor produces one set of measurements for a particular region within a time period. In a social network, a community is a set of people. It is important to devise algorithms that can effectively process and learn from these data.

A convenient and often adopted way to deal with point sets is to construct a feature vector for each set so that standard learning techniques can be applied. However, this conversion process often relies on human effort and domain expertise and is prone to information loss. Recently, several algorithms were proposed to directly learn from point sets based on the assumption that each set is a sample from an underlying distribution. (Póczos et al., 2011, 2012) proposed novel kernels between point sets based on efficient and consistent divergence estimators. (Gretton et al., 2007; Muandet et al., 2012) designed a class of set kernels based on the kernel embedding of distributions. (Boiman et al., 2008; McCann and Lowe, 2012) developed simple classifiers for point sets based on divergences between the sets and the classes. Some parametric methods have also been proposed (Jaakkola and Haussler, 1998; Jebara et al., 2004). These methods achieved impressive empirical successes, thus showing the advantage of learning directly from point sets.

One factor that can significantly affect the effectiveness of learning is sampling bias. Sampling bias comes from the way we collect points from the underlying distributions, and makes the observed sample not representative of the true distribution. It undermines the fundamental validity of learning because the points are no longer iid samples from a distribution conditioned only on the object's type. Though it has been extensively studied in statistics, this key problem has been largely ignored by the previous research on learning from sets. The goal of this paper is to alleviate the impact of sampling bias when measuring similarities between point sets.

We consider point sets with the following structure. Let each point be described by two groups of random variables: the independent variables (*i.v.*) and dependent variables (*d.v.*). A point is collected by first specifying the value of the *i.v.*, and then observing a sample from the distribution of the *d.v.* conditioned on the given *i.v.* Figure 1 shows a synthetic example where the *i.v.* is sampled uniformly, and the *d.v.* is from the Gaussian distribution whose mean is proportional to the value of *i.v.*, forming the black line-shaped point set. Many real world situations, including surveys and mobile sensing, produce point sets of this type. In patch-based image analysis, we first specify the location of the patches as the *i.v.* and then extract their features as the *d.v.* In traffic monitoring, a helicopter is sent to specific locations at specific times (*i.v.*) and measures the traffic

volume (*d.v.*).

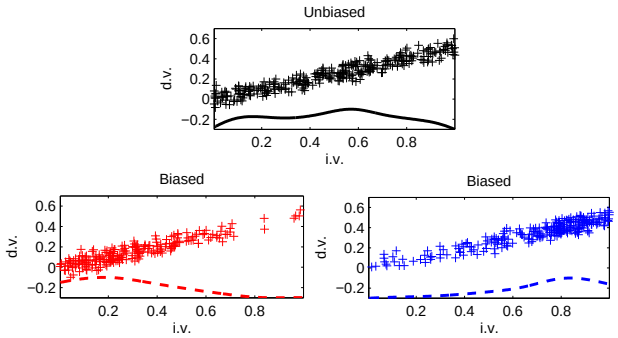


Figure 1: The observation biases.

We assume that the sampling bias affects the way we observe *i.v.* , yet the observation of *d.v.* given *i.v.* remains intact. This assumption is compatible with the covariate shift model (Shimodaira, 2000; Huang et al., 2007). As shown in Figure 1, an unbiased observer will sample *i.v.* uniformly and get the black set. Biased observers might focus more on the smaller or larger values of the *i.v.* and create the biased red and blue sets, where the curves show the observed marginal densities of the *i.v.* The joint and marginal distributions of the biased sets now look very different from each other and the unbiased set. Nevertheless, no matter what the distribution of *i.v.* is, the distribution of *d.v.* given *i.v.* is always the same Gaussian that does not change with the observer. In traffic monitoring, the helicopter may be tasked with other, non-traffic, jobs that create different patrol schedules each day, thus creating an uneven profile of the city's traffic. But the measured traffic volumes at the patrolled locations are still accurate.

To correct sampling biases of this kind, we propose to use conditional divergences. Existing divergence-based methods use the joint distribution of the *i.v.* and the *d.v.* to measure the differences between point sets. On the other hand, conditional divergences focus on the conditional distributions of *d.v.* given *i.v.* and are insensitive to the distribution of *i.v.*, which is distorted by the sampling bias in our setting. As long as the conditional distributions are intact, the conditional divergences will be reliable. Moreover, it can be shown that the divergence between joint distributions is a special case of the conditional divergence. A fast and consistent estimator is developed for the conditional divergences. We also discuss specific examples of correcting sampling biases, including some extreme cases.

We evaluate the effectiveness of conditional divergences on both synthetic and real world data sets. On synthetic data sets, we show that the proposed estimator is accurate and the conditional divergences are capable of correcting sampling biases. We also demonstrate their performance on real-world climate and image classification problems.

The rest of this paper is organized as follows. The background and some related work is introduced in Section 2. Section 3 defines the conditional divergence and describes its properties and estimation. Section 4 describes how to use conditional divergence to correct various sampling biases. In Section 5 we make a discussion about the conditional divergences. In Section 6, we evaluate the effectiveness of the proposed methods on both synthetic and real data sets. We conclude the paper in Section 7.

2 BACKGROUND AND RELATED WORK

We consider a data set that consists of M point sets $\{G_m\}_{m=1,\dots,M}$, and each point set G_m is a set of d-dimensional vectors, $G_m = \{z_{mn}\}_{n=1,\dots,N_m}$, $z_{mn} \in \mathbb{R}^d$. Each point $z_{mn} = [x_{mn}; y_{mn}]$ is a concatenation of two shorter vectors $x_{mn} \in \mathbb{R}^{d_x}$ and $y_{mn} \in \mathbb{R}^{d_y}$ representing the independent variables *i.v.* and the dependent variables *d.v.* respectively. We assume that each G_m has an underlying distribution $f_m(z) = f_m(x, y)$, and the points $\{z_{mn}\}$ are *i.i.d.* samples from $f_m(z)$. f_m can be written as $f_m(z) = f_m(y|x)f_m(x)$. In the context of image classification, each G_m is an image, and x_{mn} is the location of the nth patch and y_{mn} is the feature of that patch.

We can learn from these sets by estimating the divergence between the f_m's as the dissimilarity between the G_m's. Having the dissimilarities, various problems can be solved by using similarity based learning algorithms, including *k-nearest neighbors* (KNN), *spectral clustering* (Ng et al., 2001), and *support vector machines* (SVM). In this direction, several divergence-based methods have been proposed (Boiman et al., 2008; Póczos et al., 2012; Muandet et al., 2012), and both empirical and theoretical successes were achieved.

In the presence of sampling bias that affects the distribution of *i.v.*, $f_m(x)$ is transformed into $f'_m(x)$. Consequently the observed G_ms represent the biased joint distribution $f'_m(z) = f_m(y|x)f'_m(x)$. Therefore naïvely learning from the point sets using joint distributions will lead us to the distorted f'_m's instead of the true f_m's. To correct the sampling bias, we need to either 1) modify the point sets to restore $f(z)$, or 2) use similarity measures that are insensitive to $f(x)$.

Existing correction methods often reweigh the points in the training set so that its effective distribution matches the distribution in the test set (Shimodaira, 2000; Huang et al., 2007; Cortes et al., 2008). Our proposed conditional divergences are insensitive to the biased distributions of the independent variables and thus robust against sampling biases.

Traditionally in statistics and machine learning, sampling bias is considered between the training set and the test set. In contrast, we consider problems consisting of a large

number of point sets, and our goal is to learn from the sets themselves. This extension raises many important challenges, including how to find a common basis to compare all pairs of distributions, how to deal with unobserved segments of distributions, and how to design efficient algorithms.

To our knowledge, this is first time sampling bias is addressed in the context of learning from sets of points. Algorithms such as (Póczos et al., 2011, 2012; Gretton et al., 2007; Muandet et al., 2012; Boiman et al., 2008; McCann and Lowe, 2012; Jebara et al., 2004) all directly compare the joint distributions of the observed points, making them susceptible to sample bias. (Póczos, 2012) proposed the use of conditional divergence, yet sampling bias was still not considered.

3 CONDITIONAL DIVERGENCES

We propose to measure the dissimilarity between two distributions $p(z) = p(x, y)$ and $q(z) = q(x, y)$ using the *conditional divergence* (**CD**) based on the *Kullback-Leibler* (KL) divergence:

$$\mathrm{CD}_{c(x)}\left(p(z)||q(z)\right) = \mathbb{E}_{c(x)}\left[\mathrm{KL}\left(p(y|x)||q(y|x)\right)\right] \quad (1)$$

where $c(x)$ is a user-specified distribution over which the expectation is taken. CD is the average KL divergence between the conditional distributions $p(y|x)$ and $q(y|x)$ over possible values of x, and $c(x)$ can be considered as the importance of the divergences at different x's. CD's definition is free of the *i.v.* distributions $p(x)$ and $q(x)$, which are vulnerable to sampling biases. By definition, CD has a lot in common with the KL divergence: it is non-negative, and equals zero if and only if $p(y|x) = q(y|x)$ for every x within the support of $c(x)$. CD is also not a metric and not even symmetric.

In the form of (1), CD is hard to compute because the divergences $\mathrm{KL}\left(p(y|x)||q(y|x)\right)$ are not available for arbitrary continuous distributions. Also note that $c(x)$ is a distribution specified by the user. To make CD more accessible, we can rewrite it as

$$\begin{aligned}&\mathrm{CD}_{c(x)}\left(p(z)||q(z)\right) \quad (2)\\ &= \mathbb{E}_{p(z)}\left[\frac{c(x)}{p(x)}\left(\ln\frac{p(z)}{q(z)} - \ln\frac{p(x)}{q(x)}\right)\right].\end{aligned}$$

Now, CD is defined in terms of the density ratios of the input distributions and the expectation over $p(z)$.

An interesting case of (2) occurs when we choose $c(x) = p(x)$, which gives the result

$$\begin{aligned}&\mathrm{CD}_{p(x)}\left(p(z)||q(z)\right) \quad (3)\\ &= \mathrm{KL}(p(z)||q(z)) - \mathrm{KL}(p(x)||q(x)).\end{aligned}$$

We can see this special CD is equal to the *joint divergence* (divergence between joint distributions) minus the divergence between the marginal distributions of x. Intuitively, CD is removing the effect of $p(x)$ and $q(x)$ from the joint divergence, so that the net results are free from the sampling bias. Moreover, when $p(x)$ and $q(x)$ are the same, $\mathrm{KL}(p(x)||q(x))$ vanishes and this CD equals the joint divergence. In other words, when there is no sampling bias, $\mathrm{CD}_{p(x)}\left(p(z)||q(z)\right) = \mathrm{KL}(p(z)||q(z))$.

3.1 ESTIMATION

In this section we give an estimator for CD (2). Suppose we have two sets G_p and G_q with underlying distributions $p(z)$ and $q(z)$ respectively. We can approximate the expectation (2) with the empirical mean and estimated densities:

$$\begin{aligned}&\widehat{\mathrm{CD}}_{c(x)}\left(p(z)||q(z)\right) \quad (4)\\ &= \frac{1}{N_p}\sum_{n=1}^{N_p}\frac{c(x_{p,n})}{\hat{p}(x_{p,n})}\left(\ln\frac{\hat{p}(z_{p,n})}{\hat{q}(z_{p,n})} - \ln\frac{\hat{p}(x_{p,n})}{\hat{q}(x_{p,n})}\right),\end{aligned}$$

where N_p is the size of G_p, $\hat{p}, \hat{q}$ are the estimates of p, q.

$c(t)$ is an arbitrary input from the user and we can see that its role is to reweight the log-density-ratios at different points in G_p. To generalize this notion, we define the *generalized conditional divergence* (GCD) and its estimator as the weighted average of the log-density-ratios:

$$\begin{aligned}&\mathrm{GCD}_w\left(p(z)||q(z)\right) \quad (5)\\ &= \sum_{n=1}^{N_p} w(x_{p,n})\left(\ln\frac{p(z_{p,n})}{q(z_{p,n})} - \ln\frac{p(x_{p,n})}{q(x_{p,n})}\right)\\ &\widehat{\mathrm{GCD}}_w\left(p(z)||q(z)\right) \quad (6)\\ &= \sum_{n=1}^{N_p} w(x_{p,n})\left(\ln\frac{\hat{p}(z_{p,n})}{\hat{q}(z_{p,n})} - \ln\frac{\hat{p}(x_{p,n})}{\hat{q}(x_{p,n})}\right)\\ &\sum_{n=1}^{N_p} w(x_{p,n}) = 1, w(x_{p,n}) \geq 0,\end{aligned}$$

where $w(x)$ is the weight function and the constraint $\sum_n w(x_n) = 1$ is induced by the fact that

$$\begin{aligned}\lim_{N_p\to\infty}\sum_{n=1}^{N_p} w(x_{p,n}) &= \lim_{N_p\to\infty}\frac{1}{N_p}\sum_{n=1}^{N_p}\frac{c(x_{p,n})}{p(x_{p,n})}\\ &= \mathbb{E}_{p(x)}\left[\frac{c(x)}{p(x)}\right] = \int\frac{c(x)}{p(x)}p(x)dx = 1.\end{aligned}$$

To obtain the density estimates $\hat{p}, \hat{q}$, we use the *k-nearest-neighbor* (KNN) based estimator (Loftsgaarden and Quesenberry, 1965). Let the $f(z)$ be the d-dimensional density function to be estimated and $Z = \{z_n\}_{n=1,\ldots,N} \in \mathbb{R}^d$ be samples from $f(z)$. Then the density estimate at the point

z' is

$$\hat{f}(z') = \frac{k}{Nc_1(d)\phi_{Z,k}^d(z')}, \tag{7}$$

where $c_1(d)$ is the volume of the unit ball in the d-dimensional space, and $\phi_{Z,k}(z')$ denotes the distance from z' to its kth nearest neighbor in Z (if z' is already in Z then it is excluded). This estimator is chosen over other options such as the *kernel density estimation* because it is simple, fast, and leads to a provably convergent estimator as shown below.

By plugging in (7) into (6), we can get the following estimator for GCD:

$$\widehat{\mathrm{GCD}}_w\left(p(z)||q(z)\right) \tag{8}$$
$$= \sum_{n=1}^{N_p} w(x_{p,n}) \left(d \ln \frac{\phi_{G_q,k}(z_{p,n})}{\phi_{G_p,k}(z_{p,n})} - d_x \ln \frac{\phi_{G_q,k}(x_{p,n})}{\phi_{G_p,k}(x_{p,n})} \right),$$

where d_x is the dimensionality of the x. We can see that the resulting estimator has a simple form and can be calculated based only on the KNN statistics ϕ, which are efficient to compute using space-dividing trees or even approximate KNN algorithms such as (Muja and Lowe, 2009). Also note that even though the estimator (8) is obtained using the density estimator (7), its final form only involves simple combinations of the log-KNN-statistics $\ln \phi$. Thus, this GCD estimator effectively avoids explicit density estimation which is notoriously difficult, especially in high dimensions.

More importantly, the GCD estimator (8) has stronger convergence properties than the density estimator from which it is derived. Standard convergence results have that the density estimator (7) is statistically consistent only if $k/n \to 0, k \to \infty$ simultaneously. However, for estimator (8) convergence can be achieved even for a fixed finite k. This means that we can always use a small k to keep the nearest neighbor search fast and still get good estimates. Specifically, following the work of (Wang et al., 2009; Póczos and Schneider, 2011), the following theorem can be proved:

Theorem 1. *Suppose the density function pairs* $(p(z), q(z))$ *and* $(p(x), q(x))$ *are both* 2*-regular (as defined in (Wang et al., 2009)). Also suppose that the weight function satisfies* $\lim_{N_p \to \infty} w(x_{p,n}) = 0, \forall n$. *Then the estimator* (8) *is* L^2 *consistent for any fixed* k. *That is*

$$\lim_{N_p, N_q \to \infty} \mathbb{E}\left[\widehat{GCD}_w(p(z)||q(z)) - GCD_w(p(z)||q(z))\right]^2 = 0$$

The proof of Theorem 1 is similar to what was used in (Wang et al., 2009). The condition $\lim_{N_p \to \infty} w(x_{p,n}) = 0$ ensures that the weight function does not concentrate on only a few points. We omit the detailed proof here. Note that the convergence of GCD does not carry to CD (4) because the weight function $w(x_{p,n}) = \frac{c(x_{p,n})}{\hat{p}(x_{p,n})}$ is no longer deterministic. However, empirically we found that (4) exhibits the behavior of a consistent estimator and produces satisfactory results.

4 CHOOSING $c(x)$

To use CD, we have to choose the appropriate $c(x)$ or $w(x)$. When learning from point sets, it is preferable to use the same $c(x)$ to compute the CDs between all pairs of sets, so that they have a common basis to compare. However, this is not always necessary or possible. Even though the choice of $c(x)$ and $w(x)$ can be arbitrary, we consider 3 options below.

First, we can let $c(x) \propto 1$ so that $w(x_{p,n}) \propto p^{-1}(x_{p,n})$ to treat every value of x equally. The disadvantage is that $p^{-1}(x_{p,n})$ has to be estimated, which is error prone. We can also use $c(x) = p(x)$ and $w(x_{p,n}) \propto 1$, leading to (3). In this case, different pairs of sets can have different $c(x)$'s. When the sampling bias is small, these differences might be acceptable considering the possible errors in $w(x)$ otherwise. Thirdly, $c(x) \propto p(x)q(x)$ and $w(x_{p,n}) \propto q(x_{p,n})$ puts the focus on regions where both $p(x)$ and $q(x)$ are high. It means that we should put larger weights in dense regions and avoid scarce regions to get reliable estimates.

One caveat is that the weight function and the log-density-ratios in CD should not use the same density estimate, otherwise the estimation errors will correlate and cause systematic overestimations. Using different estimators can help decouple the errors and avoid accumulation. In practice, we use the estimator (7) with a different k.

Some extreme cases of sampling biases are when whole segments of the distribution are missing from the sample and therefore unobserved. Two sets can even have disjoint supports of x. With the CD, we can choose $c(x) \propto p(x)q(x)$ or $c(x) \propto I(p(x)q(x) > 0)$, where $I(\cdot)$ is the indicator function, and only compare two sets in their overlapping regions. The resulting quantity may not be accurate with respect to the true unbiased divergence, but it is still a valid measurement of the differences between conditional distributions. When $f(y|x)$ only weakly depends on x, this estimate can be an adequate approximation to the original divergence. If $f(y|x)$ varies drastically for different x's without any regularity then only comparing the overlapping regions might be the best we can do.

When two sets have disjoint supports in x, no useful information can be extracted and the corresponding divergence has to be regarded as missing without further assumptions. Nevertheless, in our settings where a large number of point sets are available, it is likely that each set will share its support in x with at least some others to provide a few reliable divergence estimates. We might be able to infer the diver-

gence between disjoint sets using the idea of triangulation. We shall leave this possibility for future investigation.

5 DISCUSSION

In CD, $c(x)$ conveys prior knowledge about the importance of different x's. It should be carefully chosen based on the data, and poor results can happen when the assumptions made in $c(x)$ are not valid. For example, $c(x) \propto 1$ assumes that all the x's are equally important. This could be a bad assumption when the supports of two sets do not overlap, because at some x's one of the densities will be zero, making the conditional densities $f(y|x)$ not well-defined. Similar problems might occur in regions where one of the densities is very low. Numerically the estimator can still work but usually produces poor results. In this scenario, $c(x) \propto p(x)q(x)$ suits the data better.

The CD estimator (8) relies on the KNN statistics ϕ which is the distance between nearest neighbors. Usually we use Euclidean distance to measure the difference between points and find nearest neighbors. However, the estimator does not prevent the use of other distances. In fact, (Loftsgaarden and Quesenberry, 1965) shows that alternative distances can be used and the consistency results will generally still hold. A common choice of adaptive distance measure is the *Mahalanobis distance* (Bishop, 2007), which is equivalent to applying a linear transformation to the random variables. It is even possible to learn the distance metric for ϕ in a supervised way to maximize the learning performance. We leave this possibility as future work.

The estimated conditional divergences can be used in many learning algorithms to accomplish various tasks. In this paper, we use kernel machines to classify point sets as in (Póczos et al., 2011, 2012). Having the divergence estimates, we convert them into Gaussian kernels and then use SVM for classification. When constructing kernels, all the divergences are symmetrized by taking the average $\mu(p,q) = \frac{d(p||q)+d(q||p)}{2}$. The symmetrized divergences μ are then exponentiated to get the Gaussian kernel $k(p,q) = \exp\left(-\gamma\mu(p,q)\right)$ and the kernel matrix $\mathbf{K}$, where γ is the width parameter. Unfortunately, $\mathbf{K}$ usually does not represent a valid Mercer kernel because the divergence is not a metric and random estimation errors exist. As a remedy, we discard the negative eigenvalues from the kernel matrix $\mathbf{K}$ to convert it to its closest *positive semi-definite* (PSD) matrix $\tilde{\mathbf{K}}$. This $\tilde{\mathbf{K}}$ then is a valid kernel matrix and can be used in an SVM for learning.

6 EXPERIMENTS

We examine the empirical properties of the conditional divergences and their estimators. The tested divergences are listed below.

- **Full D**: Divergence between full unbiased sets as the groundtruth.
- **D**: Divergence between biased sets.
- **D-DV**: Divergence between biased sets while ignoring the *i.v.*.
- **CD-P,CD-U,CD-PQ**: conditional divergences with $c(x) \propto p(x), c(x) \propto 1, c(x) \propto p(x)q(x)$ respectively between biased sets.

Full D, D, D-DV are estimated using the KL divergence estimator proposed by (Wang et al., 2009). Unless stated otherwise, we use $k = 3$ for GCD estimation using (8), and use k values between 30 and 50 to compute the weight function.

We consider two types of sampling biases. The first type creates different $f(x)$'s for different sets, yet they still share the same support of x as the original unbiased data. Based on the first type, the second type of sampling bias is more extreme and can hide certain segments of the true distributions, and thus causes different sets to have different supports of x. We call the resulting test sets from these two sampling biases *uneven sets* and *partial sets* respectively.

In order to evaluate the quality of the bias correction by the CDs, we use controlled sampling biases in our experiments. The original point set data are collected from real problems without any sampling bias. Then we resample each set to create artificial sampling biases. By doing this, we can compare the results using the biased sets to the divergences using the unbiased data which is the groundtruth.

An SVM is used to classify the point sets using the method described in Section 5. When using the SVM, we tune the width parameter γ and the slack penalty C by 3-fold cross-validation on the training set.

6.1 SYNTHETIC DATA

6.1.1 Estimation Accuracy

We generate synthetic data to test the accuracy of the proposed conditional divergence estimators. The data set consists of 2-dimensional (one as *i.v.* and one as *d.v.*) Gaussian noise along two horizontal lines as the two classes, as shown in Figure 2 and 3. The Gaussians have fixed spherical covariance, and the mean of the blue class is slightly higher than the red class, resulting in an analytical KL divergence of 0.5. Then the *i.v.* (x axis) is resampled to create sampling bias and the red and blue curves show the resulting marginal densities $f_{\text{red}}(x), f_{\text{blue}}(x)$. The task is to recover the true divergence value 0.5 from this biased sample. We vary the sample size to see the empirical convergence, and the results of 10 random runs are reported. The shortcut for this problem is to ignore the *i.v.*, but we do not let the estimators take it and force them to recover from the sampling bias.

Figure 2 shows the results on the uneven sets. As expected, the joint divergences are corrupted by the sampling bias and are far from the truth. The three CDs all converge to the true value. Figure 3 shows the results on the partial sets. The joint divergence diverges in this case. CD-P and CD-U are closer but not converging to the correct value, and the reason is that the non-overlapping supports violate the assumptions made by them. CD-PQ successfully achieved the true value. This shows the advantage of only measuring CD within the overlapping region in this example. Overall, the CDs are effective against sampling bias and the estimators converge to the true values.

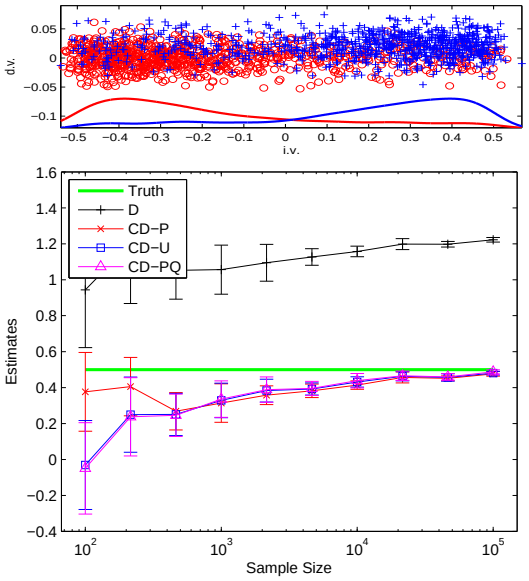


Figure 2: Divergences on the uneven synthetic data.

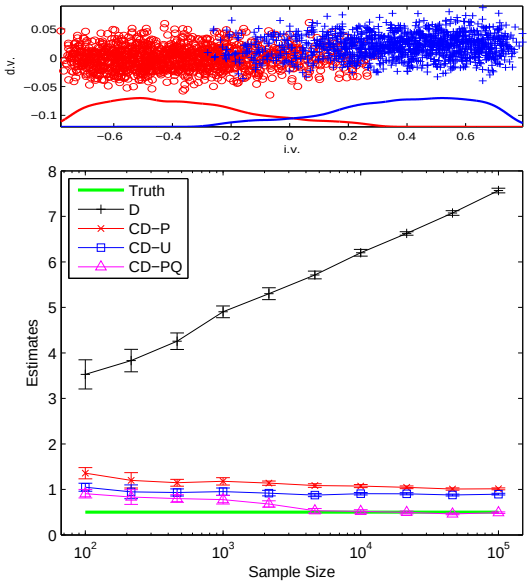
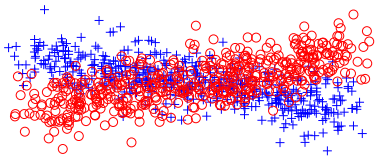


Figure 3: Divergences on the partial synthetic data.

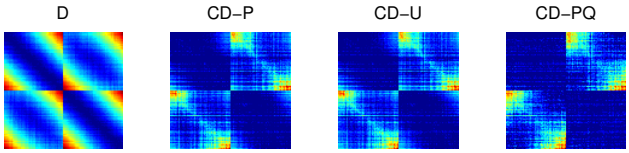
6.1.2 Handling Point Sets

Here we test the estimators using a large number of point sets. The full data of two classes are shown in Figure 5a. To create partial sets, we use a sliding window, whose width is half of the data's span, to scan the full data and at each position put the points within the window together as a set. The uneven sets are then created by combining the partial sets with a small number of random samples from the original data. 100 sets are created for each class and each set contains $200 - 300$ points.

This data set is more challenging: the marginal distribution of *d.v.* cannot differentiate the two classes; the conditional distributions $f(y|x)$ are dependent on x; near the center of the data the conditional distributions of the two classes are very close. The different divergence matrices on the uneven sets are shown in Figure 4, in which we sorted the sets according to their classes and window positions to show the structures. We see that the joint divergence is severely affected by the sampling bias, while the CDs are quite insensitive. The result of CD-U is especially impressive: the similarity structure of the original data is perfectly recovered. Figure 5 shows the results on the partial sets. The joint divergence is now dominated by the sampling bias. CDs again are able to recover from this severe disruption and achieve reasonable results. The result of CD-PQ is the cleanest on this data set.



(a) Original data.



(b) Divergences

Figure 5: Divergences on the partial sets. The goal is to recover the "Full D" result shown in Figure 4.

6.2 SEASON CLASSIFICATION

In this section we use the divergences in SVM to classify real world point sets generated by sensor networks. We gathered the data from the QCLCD climate database at NCDC [1]. We use a subset of QCLCD that contains daily climatological data from May 2007 to May 2013 measured by $1,164$ weather stations in the continental U.S. Each of these weather station produces various measurements such

[1] http://www.ncdc.noaa.gov

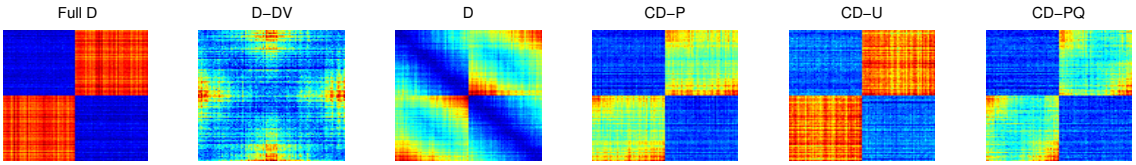
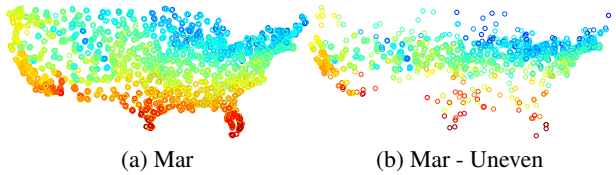


Figure 4: Divergences on the uneven sets. The goal is to recover the "Full D" given only the biased sets.

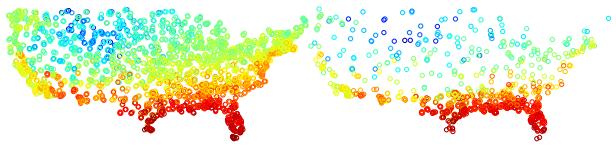
as the temperature, humidity, precipitation, *etc*, at its location. We aggregate these data into point sets, so that each set contains the measurements from all stations in one week.

We consider the problem of predicting the season of a set based on the average temperature measurement. Specifically, we want to know if a set corresponds to Spring or Fall based on the average temperatures over the U.S. Note that classifying Summer and Winter would be too easy, while differentiating Spring and Fall can be challenging since they have similar average temperatures. Nevertheless, it is still possible based on the geographical distribution of the temperatures. Figure 6 shows the temperature maps in a first week of March and a first week of November.

Again, we create uneven and partial sets based on the original data by randomly positioning a full-width window whose height is 20% of the data's vertical span, as shown in Figure 6. This injection of sampling bias is simulating the scenario where we only have a sensoring satellite orbiting parallel to the equator. In this problem, the location is the *i.v.* and the temperature is the *d.v.*. This procedure gives us 160 3-dimensional (latitude, longitude, temperature) point sets with sizes around $2,000$.



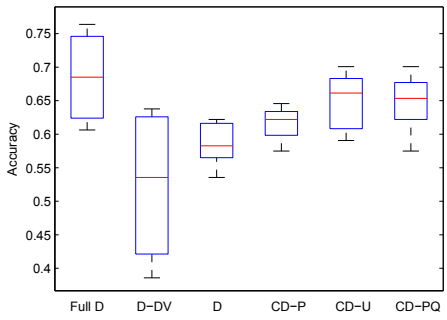
(a) Mar (b) Mar - Uneven



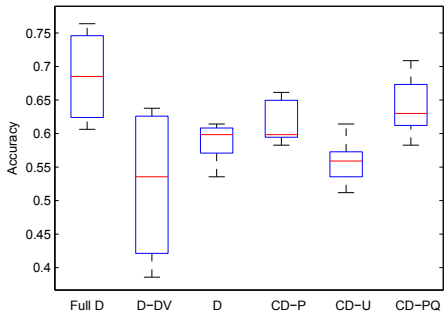
(c) Nov (d) Nov - Uneven

Figure 6: Example temperature maps of the U.S. from the QCLCD. (a) and (c) are the original data. (b) and (d) are the artificially created uneven data.

In each run, 20% of the random point sets are used for training and the rest are used for testing. Classification results of 10 runs are reported in Figure 7. On the uneven sets, we see that both CD-U and CD-PQ are able to recover from the sampling bias and achieve results that are only 3% worse than the full divergence. On the partial sets, however, the performance CD-U dropped significantly. This indicates that it can be risky to apply CD in regions where two sets do not overlap. It is interesting to see that D-DV, which ignores the locations, barely does better than random since Spring and Fall indeed have similar temperatures. Yet by considering the geographical distribution of temperatures we can achieve 70% accuracy.



(a) QCLCD, uneven.



(b) QCLCD, partial.

Figure 7: Season classification results on the QCLCD weather data.

6.3 IMAGE CLASSIFICATION

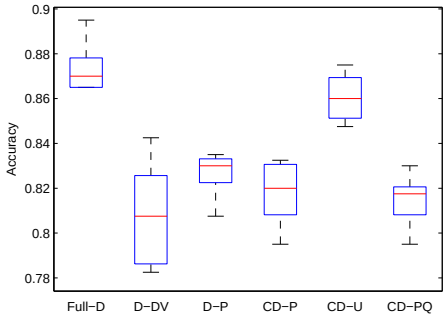
We can also use CDs to classify scene images. We construct one point set for each image, where each point describes one patch including its location (*i.v.*) and the feature (*d.v.*). The OT (A.Oliva and Torralba, 2001) scene images are used, which contain $2,688$ grayscale images of size 256×256 from 8 categories. The patches are sampled densely on a grid and multiscale SIFT features are extracted using VLFeat (Vedaldi and Fulkerson, 2008). The points are reduced to 20-dimensions using PCA, preserving 70%

of variance.

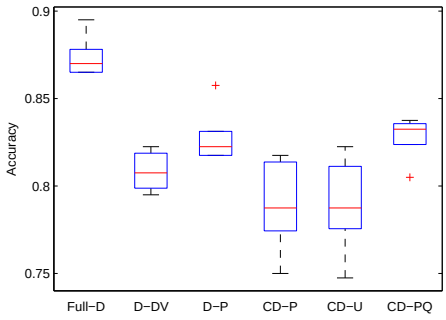
Again, we create both uneven and partial point sets by randomly positioning a full-width window whose height is 60% of the image. By doing this, the injected sampling bias forces a set to focus on a specific horizontal part of the scene. For instance in a beach scene, the biased observer focuses either on the sky or the sand, and only see a small part of the rest of the scene. After the above processing, the full data set contains $2,688$ sets of 20-dimensional points, and the sets' sizes are around $1,600$. In the biased data, each partial set has about 950 points and each uneven set has about $1,100$. In each run, we randomly select 50 images per class for training and another 50 for testing.

Results of 10 random runs are shown in Figure 8. In these results, CDs again successfully restore the accuracies to a high level even in the face of harsh sampling biases. We see that CD-U impressively beats the other methods by a large margin on the uneven sets, and is only 1% worse than the full divergence. CD-PQ is the best on partial sets. These results show the CDs' corrective power when the correct assumptions are made about the sampling biases.

We also observe that CD-U and CD-P did not perform well on the partial sets, which is expected since their assumptions were invalid on the data. In general, the impact of sampling bias on this data set is small (less than 10% decrease in accuracies) because the patch features (*d.v.*) only weakly depend on the patch locations (*i.v.*). In fact, many patch-based image analyses such as (Li and Perona, 2005) do not include the locations. This might explain why both D-DV and D-P did reasonably well in this task and the corrected results by CD-PQ are only slightly better.



(a) Image, uneven.



(b) Image, partial.

Figure 8: Image classification results on OT.

7 CONCLUSION

In this paper we described various aspects of dealing with sampling bias when learning from point sets. We proposed the conditional divergence (CD) to measure the difference between point sets and alleviate the impact of sampling bias. An efficient and convergent estimator of CD was provided. We then discussed how to deal with various types of sampling biases using CD. In the experiments we show that these methods are effective against sampling bias on both synthetic and real data.

Several directions can be explored in the future. We can extend the definition of conditional divergence from KL divergence to the more general Rényi divergences. The generalized conditional divergences provide the possibility of learning the weights of the density ratios in a supervised ways in order to maximize the discriminative power of the resulting divergences. The distance between points used in estimating the CDs could also be learned. Finally for extreme cases that cause missing divergences, we may infer them by exploiting the relationships among the sets using matrix completion techniques.

Acknowledgments

This research is supported by DARPA grant #FA87501220324.

References

A.Oliva and A. Torralba. Modmodel the shape of the scene: a holistic representation of spatial envelope. *International Journal of Computer Vision (IJCV)*, 42, 2001.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

Oren Boiman, Eli Shechtman, and Michal Irani. In defense of nearest neighbor based image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In *Algorithmic Learning Theory*, 2008.

Arthur Gretton, Karsten M. Borgwardt, Malte Rasch, Bernard Schölkopf, and Alex J. Smola. A kernel method for the two sample problem. In *Neural Information Processing Systems (NIPS)*, 2007.

Jiayuan Huang, Alexander J. Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Schölkopf. Cor-

recting sample selection bias by unlabeled data. In *NIPS*, 2007.

Tommi Jaakkola and David Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1998.

T. Jebara, R. Kondor, A. Howard, K. Bennett, and N. Cesabianchi. Probability product kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

Fei-Fei Li and Pietro Perona. A bayesian hierarchical model for learning natural scene categories. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2005.

D. O. Loftsgaarden and C. P. Quesenberry. A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3), 1965.

Sancho McCann and David G. Lowe. Local naive bayes nearest neighbor for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Krikamol Muandet, Kenji Fukumizu, Francesco Dinuzzo, and Bernhard Schölkopf. Learning from distributions via support measure machines. In *Neural Information Processing Systems (NIPS)*, 2012.

Marius Muja and David G. Lowe. Fast approximate nearest nneighbor with automatic algorithms configuration. In *International Conference on Computer Vision THeory and Applications (VISAPP)*, 2009.

Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, 2001.

Barnabás Póczos. Nonparametric estimation of conditional information and divergences. In *AI and Statistics (AISTATS)*, 2012.

Barnabás Póczos and Jeff Schneider. On the estimation of alpha divergence. In *AI and Statistics (AISTATS)*, 2011.

Barnabás Póczos, Liang Xiong, and Jeff Schneider. Nonparametric divergence estimation with applications to machine learning on distributions. In *Uncertainty in Artificial Intelligence (UAI)*, 2011.

Barnabás Póczos, Liang Xiong, Dougal Sutherland, and Jeff Schneider. Nonparametric kernel estimators for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2), 2000.

A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

Qing Wang, Sanjeev R. Kulkarni, and Sergio Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *IEEE Trans. on Information Theory*, 55, 2009.